AMENDMENTS TO THE CLAIMS

Please cancel claims 1-16 and 22-32 without prejudice. Kindly add new claims 33-47 as shown in the following listing of claims. The listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims

1-16. (canceled)

17. (original) An apparatus for improving performance in a pipeline microprocessor having an architected register file, the apparatus comprising:

early execution logic, configured to generate early results of instructions prior to generation of final results of said instructions by one or more execution units lower in the microprocessor pipeline than said early execution logic;

an early register file, coupled to said early execution logic, comprising registers corresponding to the architected register file for storing said early results, having a valid indicator associated with each of said registers, configured to provide said early results to said early execution logic as operands for generating subsequent said early results; and

logic, coupled to receive said valid indicators, configured to stall the pipeline if said early execution logic is generating an address using an operand stored in one of said early register file registers and said valid indicators indicate that said register is invalid, wherein said logic is further configured not to stall the pipeline if said early execution logic is generating a non-memory address early result using an operand stored in one of said early register file registers and said valid indicators indicate that said register is invalid.

18. (original) The apparatus of claim 17, wherein said early execution logic is within an address stage of the pipeline microprocessor.

19. (original) The apparatus of claim 17, further comprising:

a register, coupled to said logic, configured to accumulate early status flags based on said early results, thereby enabling early execution of conditional instructions based on said early status flags accumulated in said register.

20. (original) The apparatus of claim 17, further comprising:

a result bus, coupling an output of said early execution logic to a first input of a multiplexer, for providing said early results from said early execution logic, said multiplexer having a second input coupled to receive an operand from the architected register file, said multiplexer having an output for providing at least one operand selected from at least one of said inputs to said early execution logic, whereby a first early result generated by said early execution logic during a first clock cycle is provided as an

input operand to a second instruction for generating a second early result of said second instruction during a second clock cycle immediately subsequent to said first clock cycle.

21. (original) The apparatus of claim 17, wherein the pipeline microprocessor is a scalar microprocessor.

22-32. (canceled)

33. (new) A pipeline microprocessor, comprising:

an architected register file;

one or more execution units, coupled to said architected register file;

early execution logic, coupled to said architected register file, configured to generate early results of instructions prior to generation of final results of said instructions by said one or more execution units lower in the microprocessor pipeline than said early execution logic;

an early register file, coupled to said early execution logic, comprising registers corresponding to the architected register file for storing said early results, having a valid indicator associated with each of said registers, configured to provide said early results to said early execution logic as operands for generating subsequent said early results; and

logic, coupled to receive said valid indicators, configured to stall the pipeline if said early execution logic is generating an address using an operand stored in one of said early register file registers and said valid indicators indicate that said register is invalid, wherein said logic is further configured not to stall the pipeline if said early execution logic is generating a non-memory address early result using an operand stored in one of said early register file registers and said valid indicators indicate that said register is invalid.

34. (new) The microprocessor of claim 33, wherein said early execution logic is within an address stage of the pipeline microprocessor.

35. (new) The microprocessor of claim 33, further comprising:

a register, coupled to said logic, configured to accumulate early status flags based on said early results, thereby enabling early execution of conditional instructions based on said early status flags accumulated in said register.

36. (new) The microprocessor of claim 33, further comprising:

a result bus, coupling an output of said early execution logic to a first input of a multiplexer, for providing said early results from said early execution logic, said multiplexer having a second input coupled to receive an operand from the architected register file, said multiplexer having an output for providing at least one operand selected from at least one of said inputs to said early execution logic, whereby a first early result generated by said early execution logic during a first clock cycle is provided as an input operand to a second instruction for generating a second early result

of said second instruction during a second clock cycle immediately subsequent to said first clock cycle.

37. (new) The microprocessor of claim 33, wherein the pipeline microprocessor is a scalar microprocessor.

38. (new) A method for improving performance in a pipeline microprocessor having an architected register file, the apparatus comprising:

generating early results, by early execution logic, of instructions prior to generation of final results of said instructions by one or more execution units lower in the microprocessor pipeline than said early execution logic;

storing said early results in an early register file comprising registers corresponding to the architected register file having a valid indicator associated with each of said registers;

providing said early results to said early execution logic as operands for generating subsequent said early results; and

stalling the pipeline if said early execution logic is generating an address using an operand stored in one of said early register file registers and said valid indicators indicate that said register is invalid; and

foregoing stalling the pipeline if said early execution logic is generating a non-memory address early result using an operand stored in one of said early register file registers and said valid indicators indicate that said register is invalid.

39. (new) The method of claim 38, wherein said early execution logic generates said early results within an address stage of the pipeline microprocessor.

40. (new) The method of claim 38, further comprising:

accumulating in a register early status flags based on said early results; and

performing early execution of conditional instructions based on said early status flags accumulated in said register.

41. (new) The method of claim 38, further comprising:

providing, by a result bus, from an output of said early execution logic to a first input of a multiplexer, said early results from said early execution logic;

receiving, by a second input of said multiplexer, an operand from the architected register file;

providing to said early execution logic, by said multiplexer, at least one operand selected from at least one of said inputs; and

providing a first early result generated by said early execution logic during a first clock cycle as an input operand to a second instruction for generating a second early result of said second instruction during a second clock cycle immediately subsequent to said first clock cycle.

42. (new) The method of claim 38, wherein the pipeline microprocessor is a scalar microprocessor.

43. (new) A computer program embodied on a computer-readable medium, comprising:

computer-readable program code for providing an apparatus for improving performance in a pipeline microprocessor having an architected register file, said program code comprising:

first program code for providing early execution logic, configured to generate early results of instructions prior to generation of final results of said instructions by one or more execution units lower in the microprocessor pipeline than said early execution logic;

second program code for providing an early register file, coupled to said early execution logic, comprising registers corresponding to the architected register file for storing said early results, having a valid indicator associated with each of said registers, configured to provide said early results to said early execution logic as operands for generating subsequent said early results; and

third program code for providing logic, coupled to receive said valid indicators, configured to stall the pipeline if said early execution logic is generating an address using an operand stored in one of said early register file registers and said valid indicators indicate that said register is invalid, wherein said logic is further configured not to stall the pipeline if said early execution logic is generating a non-memory address early result using an operand stored in one of said early register file registers and said valid indicators indicate that said register is invalid.

44. (new) The computer program of claim 43, wherein said early execution logic is within an address stage of the pipeline microprocessor.

45. (new) The computer program of claim 43, further comprising:

fourth program code for providing a register, coupled to said logic, configured to accumulate early status flags based on said early results, thereby enabling early execution of conditional instructions based on said early status flags accumulated in said register.

46. (new) The computer program of claim 43, further comprising:

fourth program code for providing a result bus, coupling an output of said early execution logic to a first input of a multiplexer, for providing said early results from said early execution logic, said multiplexer having a second input coupled to receive an operand from the architected register file, said multiplexer having an output for providing at least one operand selected from at least one of said inputs to said early execution logic, whereby a first early result generated by said early execution logic during a first

clock cycle is provided as an input operand to a second instruction for generating a second early result of said second instruction during a second clock cycle immediately subsequent to said first clock cycle.

47. (new) The computer program of claim 43, wherein the pipeline microprocessor is a scalar microprocessor.